Thesis Outline Rough Draft

This is derived from from the Generic MS Thesis Outline at http://users.csc.calpoly.edu/˜gfisher/students/generic-ms-toc.html. The italicized notes at the beginning of each section describe generically what's in that section. The normal-font text describes more specifically what will go in your thesis.

1. Introduction

One to three paragraph summary of what the thesis is about and what it contributes.

The focus is on the incremental validation of formal specifications.

1.1. Description of the Problem

One to three page summary of the problem to be solved, including motivation for why it is an interesting problem.

The fundamental problem is to provide the means to validate formal specifications in a straightforward manner, as a natural part of the software development process. The more general problem is to address the seemingly perennial difficulty of convincing software engineering students and practitioners to appreciate the utility of formal methods. *More blah, blah, blah goes here.*

1.2. Overview of the Solution

One to five page executive summary of the solution; this is basically a capsule description of the remainder of the thesis.

Main points:

- Provide a functional interpreter for a predicative specification language.
- Provide the means to execute preconditions and postconditions.
- Provide an interface to supply incremental test cases that validate (or invalidate) the conditions.

1.3. Possible other Subsections

Optional subsections that discuss such topics as goals and objectives, restrictions and limitations, or the like. In particular, if the system is not fully implemented, then this introductory section should clarify precisely what parts are implemented and what parts are not (call this the "truth in advertising" part.)

1.4. Outline of the Thesis

One paragraph summary of what the remaining chapters contain.

2. Background

Optional chapter that describes necessary technical background if it cannot safely be assumed that the reader has such background. For example, this section can define technical terms used in the remainder of the thesis. The section is optional since the next section can provide necessary background discussion if it is not too detailed.

It may be worthwhile to present an overview of formal methods here, and possibly move the "Related Work" chapter here, calling this chapter "Background and Related Work".

3. Scenario(s) of System Use

An end-user view of how the system works. The scenario(s) is(are) based on an actual concrete session where the user performs some interesting and useful work. The goal is to illustrate the major and most interesting system features, not all system features. If the system does not involve a substantial amount of end user interaction, this chapter can be more appropriately named "Examples of System Use".

This chapter will present a walk-through of a few particularly interesting examples.

4. Overall System Design

Description of system design, probably supported with appropriate design diagrams, including but not limited to: block diagram, UI-to-class relationship diagram, DFD, class message passing, class hierarchy, functional hierarchy.

This chapter presents an overview of the major system components. Since it's implemented in regular C, a UML class diagram isn't particularly useful. A block diagram of the major components would be useful, along the lines of a standard compiler/interpreter diagram. Also useful would be a high-level function diagram of the interpreter

Chapters follows that describe the specific technical details of the system that has been built. The technical chapters include design and implementation details of all the interesting technical problems that were totally or partially solved.

5. The Functional Interpreter

Describe a bit of the detailed design of the interpreter, with some brief sample code excerpts.

6. Quantifier Execution

Describe the possible ways to execute quantifiers, and focus on the particular technique used here.

7. User Interface

If you build a GUI, it'll get described here.

8. Results

Optional chapter that presents quantitative and/or qualitative results, such as the results of controlled end-user usage studies or quantitative comparative analysis of system performance. This chapter may appear earlier if the results are particularly significant or worthy of emphasis compared to the technical design aspects of the system.

I need to think about what can go here. As noted, it's optional.

9. Related Work

In-depth discussion of other research related to this thesis. This chapter may sometimes appear earlier, say as chapter 2 or 3, if there is a substantial amount of technical background that a reader may need to know before continuing with the rest of the thesis. If it does appear earlier, then this and the background chapter can be merged into one chapter called "Background and Related Work".

As noted for Chapter 2, this may move up to that chapter on "Background and Related Work".

10. Conclusions

General conclusions about the work described in the thesis, including a recap of what was accomplished.

10.1. Summary of Contributions

Enumerate the specific technical contributions of the thesis, emphasizing what is novel and/or noteworthy.

10.2. Future Work

Describe intended future uses and major challenges for the future. Emphasize work that is likely to be pursued in the near term.

Appendix A: Users Manual (if appropriate)

Appendix B: Code Listings (generally only brief excerpts, if any)

Appendix C: Other Pertinent Detailed Material